IBATIS - STORED PROCEDURES

http://www.tutorialspoint.com/ibatis/ibatis_stored_procedures.htm

This is very much possible to call a stored procedure using iBATIS configuration. To understand this chapter, first you need to understand how we create a stored procedure in MySQL.

Before procedding for this chapter you can go through MySQL Stored Procedure.

We have following EMPLOYEE table in MySQL:

```
CREATE TABLE EMPLOYEE (
   id INT NOT NULL auto_increment,
   first_name VARCHAR(20) default NULL,
   last_name  VARCHAR(20) default NULL,
   salary     INT  default NULL,
   PRIMARY KEY (id)
);
```

Let us have following stored procedure created in MySQL database.

```
DELIMITER $$

DROP PROCEDURE IF EXISTS `testdb`.`getEmp` $$
CREATE PROCEDURE `testdb`.`getEmp`
   (IN empid INT)
BEGIN
   SELECT * FROM EMPLOYEE
   WHERE ID = empid;
END $$

DELIMITER;
```

Consider EMPLOYEE table is having two records as follows:

```
mysql> select * from EMPLOYEE;
+----+------------+-----------+--------+
| id | first_name | last_name | salary |
+----+------------+-----------+--------+
|  1 | Zara       | Ali       |   5000 |
|  2 | Roma       | Ali       |   3000 |
+----+------------+-----------+--------+
2 row in set (0.00 sec)
```

Employee POJO Class:

To use stored procedure, you do need to modify Employee.java file. So let us keep it as it is in last chapter.

```
public class Employee {
  private int id;
  private String first_name;
  private String last_name;
  private int salary;

  /* Define constructors for the Employee class. */
  public Employee() {}

  public Employee(String fname, String lname, int salary) {
    this.first_name = fname;
    this.last_name = lname;
    this.salary = salary;
  }
```

```
 /* Here are the required method definitions */
  public int getId() {
    return id;
  }
  public void setId(int id) {
    this.id = id;
  }
  public String getFirstName() {
    return first_name;
  }
  public void setFirstName(String fname) {
    this.first_name = fname;
  }
  public String getLastName() {
    return last_name;
  }
  public void setlastName(String lname) {
    this.last_name = lname;
  }
  public int getSalary() {
    return salary;
  }
  public void setSalary(int salary) {
    this.salary = salary;
  }

 } /* End of Employee */
```

Employee.xml File:

Here we would modify Employee.xml file to introduce <procedure></procedure> and <parameterMap></parameterMap> tags. Here <procedure></procedure> tag would have an id which we would use in our application to call the stored procedure.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sqlMap
PUBLIC "-//ibatis.apache.org//DTD SQL Map 2.0//EN"
"http://ibatis.apache.org/dtd/sql-map-2.dtd">

<sqlMap namespace="Employee">

<!-- Perform Insert Operation -->
<insert >
   INSERT INTO EMPLOYEE(first_name, last_name, salary)
   values (#first_name#, #last_name#, #salary#)

   <selectKey resultClass="int" keyProperty="id">
      select last_insert_id() as id
   </selectKey>

</insert>

<!-- Perform Read Operation -->
<select >
   SELECT * FROM EMPLOYEE
</select>

<!-- Perform Update Operation -->
<update >
   UPDATE EMPLOYEE
   SET    first_name = #first_name#
   WHERE  id = #id#
</update>

<!-- Perform Delete Operation -->
<delete >
   DELETE FROM EMPLOYEE
   WHERE  id = #id#
</delete>
```

```
<!-- To call stored procedure. -->
<procedure
              parameterMap="getEmpInfoCall">
   { call getEmp( #acctID# ) }
</procedure>
<parameterMap  >
   <parameter property="acctID" jdbcType="INT"
   javaType="java.lang.Integer" mode="IN"/>
</parameterMap>

</sqlMap>
```

IbatisSP.java File:

This file would have application level logic to read name of the employee from the Employee table using ResultMap:

```
import com.ibatis.common.resources.Resources;
import com.ibatis.sqlmap.client.SqlMapClient;
import com.ibatis.sqlmap.client.SqlMapClientBuilder;
import java.io.*;
import java.sql.SQLException;
import java.util.*;

public class IbatisSP{
  public static void main(String[] args)
   throws IOException,SQLException{
   Reader rd = Resources.getResourceAsReader("SqlMapConfig.xml");
   SqlMapClient smc = SqlMapClientBuilder.buildSqlMapClient(rd);

   int id = 1;
   System.out.println("Going to read employee name.....");
   Employee e = (Employee)smc.queryForObject
                ("Employee.getEmpInfo", id);

   System.out.println("First Name:  " + e.getFirstName());

   System.out.println("Record name Successfully ");

  }
}
```

Compilation and Run:

Here are the steps to compile and run the above mentioned software. Make sure you have set PATH and CLASSPATH appropriately before proceeding for the compilation and execution.

- Create Employee.xml as shown above.
- Create Employee.java as shown above and compile it.
- Create IbatisSP.java as shown above and compile it.
- Execute IbatisSP binary to run the program.

You would get following result:

```
Going to read record.....
ID:  1
First Name:  Zara
Last Name:  Ali
Salary:  5000
Record read Successfully
```